4. commit_a_transaction( );

The mechanism notes the close of a transaction and notifies all currently active storage managers to commit the transaction. In response, the external storage manager for the RDB issues a "COMMIT TRANSACTION" statement to the RDB.

EXAMPLE 3

The following program would be used to purge a pitcher object from the object database.

```
1.  start_a_transaction ( );
2.  query pitcher_query ("THE pitcher WITH name =
    'John Smith'");
3.  pitcher smith = the_next_result_from (pitcher_query);
4.  delete (smith);
5.  commit_a_transaction ( );
```

Steps 1 through 3 and 5 would have the same affect as described hereinbefore in Example 2.

The code:

3. pitcher smith=the_next_result_from(pitcher_query); deletes the object from the database. As illustrated in FIG. **10**, the external storage manager mechanism **26** sends a message to the object's key **90**. The key passes the message along to the Typemap **92**.

The Typemap **92** issues the following SQL commands **94** based on the data in the key represented by ". . . ":

```
DELETE
from PITCHER
where NAME = . . .;
DELETE
from PLAYER
where NAME = . . .;
```

This effects deletion of the object from the external relational database as depicted in FIG. **10**.

The illustrative implementation, setforth hereinbefore, of an integration environment according to the invention effectively integrates an external relational data store with an ODBMS by implementing a flexible interface between the relational and object paradigms.

Although the illustrative embodiment described hereinbefore includes examples wherein data from relational databases is mapped to object applications, it should be appreciated the reverse can effectively be implemented according to the invention, and that other types of storage facilities can be accessed according to the invention including hierarchical and network databases, simple data files, other object oriented databases, or combinations of these formats.

Further, while the functionality described illustrates: purging objects created from mapped relational data; constructing iterators and RDB query cursors to implement an object query in the context of an RDB; constructing keys for the purposes of constructing objects from relational data; and updating/changing objects and relational data, among other things, it will be appreciated that additional functionality can be implemented using the functions and concepts disclosed hereinbefore.

Although iterators, as known in the art, are used in association with particular external storage managers as described herein, it should be appreciated that external storage management according to the invention can be implemented with a common iterator protocol, or dedicated iterators serving particular types of storage managers.

While the invention has been shown and described with respect to an illustrative embodiment thereof, various other changes, omissions and additions in the form and detail thereof may be made therein without departing from the spirit and scope of the invention.

What is claimed is:

1. Apparatus for integrating non-object structured data, stored in an external data store having an external data store interface, with at least one object application which processes object data structures through an ODBMS application interface, comprising:

- a storage manager mechanism responsive to said ODBMS application interface and said external data store interface, including,
- a first interface, between said ODBMS application interface and said storage manager mechanism, said first interface comprising a plurality of first interface constructs including objects, references, indices, extensions and transactions, and facilitating handling of said plurality of first interface constructs for communicating with said ODBMS application interface to transfer object data structures between said storage manager mechanism and said at least one object application;
- an integral mapping mechanism responsive to said first interface and receiving at least some of said plurality of first interface constructs, said mapping mechanism mapping said at least some of said plurality of first interface constructs to a plurality of second interface constructs to effect transformation of object data structures to non-object structured data for storage in said external data store and to effect transformation of non-object structured data to object data structures for use by said at least one object application; and
- a second interface between said mapping mechanism and said external data store, said second interface comprising said plurality of second interface constructs and facilitating handling of said plurality of second interface constructs for communicating with said external data store interface to transfer non-object structured data between said storage manager mechanism and said external data store.

2. The apparatus of claim **1** wherein said external data store is at least one relational database.

3. The apparatus of claim **1** wherein said storage manager mechanism includes a repository storing information for mapping said at least some of said plurality of first interface constructs to at least some of said plurality of second interface constructs.

4. The apparatus of claim **3** wherein said repository includes schema information describing the structure of data stored in said external data store.

5. The apparatus of claim **2** wherein said second interface communicates directly with said external data store and said plurality of second interface constructs includes a key construct which identifies a particular record in said external data store.

6. The apparatus of claim **2** wherein said second interface communicates directly with said external data store and said plurality of second interface constructs includes a typemap construct which defines the transformation between non-object structured data and object data structures.

7. The apparatus of claim **2** wherein said second interface communicates directly with said external data store and said plurality of second interface constructs includes a typemap construct which defines the transformation between non-